REMARKS/ARGUMENTS

Claims 1-115 are pending in this application. The Examiner has objected to claim 39 on the basis that it is a substantial duplicate of claim 64.

Applicant has carefully examined all the claims in this application (as he did prior to filing this application) and submits that the claims are of varying scope and further, that Applicant is entitled to make claims of varying scope. In particular, with respect to claim 39, which depends indirectly on claim 7 and thence to claim 1, Applicant notes that claim 39 recites, inter alia, (see claim 1) "the user interface and controller communicating with the data base whereby the data base supplies first and second data stored in the data base to the user interface and controller, the first data being provided to the user interface and controller for providing a format for the request to the user and the second data being provided to the user interface and controller for preparing text in a document". Accordingly, claim 39 recites that first and second data are supplied by the data base to the user interface.

Claim 64, in contrast, recites "the user interface and controller communicating with the data base whereby the data base supplies data stored in the data base to the user interface and controller, the data being provided to the user interface and controller for providing information for presenting the request to the user and inserting data into the document". Claim 64 recites that the data base supplies data to the user interface and controller and omits the recitation of first and second data. Accordingly, the scope of claim 64 is different than the scope of claim 39 and accordingly it is not a substantial duplicate of claim 39.

Applicant has reconsidered all the claims in this application and submits that none of the claims is a substantial duplicate of any other claim. Accordingly, Applicant respectfully submits that the objections to the claims should be withdrawn.

The Examiner has rejected claims 1-53 and 59-64 under 35 U.S.C. §112, first paragraph, as failing to comply with the enablement requirement. In particular, the Examiner asserts that the claims contain subject matter which is not described in the specification in such a way as to enable one skilled in the art to which it pertains or to with which it is most nearly connected to make and/or use the invention. More particularly, the Examiner asserts that claims 1, 53 and 59-64 recite "a user interface and controller having a plurality of programming functionalities" and

that the specification teaches at lines 3-5 of page 15 a programming environment, not a user interface and/or controller having programming functionalities.

At page 15, lines 3-5, it is stated that "the overall system preferably comprises a user interface and controller comprising a full programming environment 10 comprising the program to be discussed herein and having full programming functionalities capable of being accessed by the program".

Applicant submits that the recitation of "a user interface and controller having a plurality of programming functionalities" in the claims refers to the programming environment 10 described on page 15 and clearly shown in the drawings. In particular, in Fig. 1, the full programming environment 10 is also identified as a "user interface and controller" so the terminologies are clearly synonymous. Accordingly, it is submitted that the description does describe "a user interface and controller having a plurality of programming functionalities" and that therefore the rejection contained in the section numbered 5 of the Detailed Office Action based on 35 U.S.C. §112, first paragraph, should be withdrawn. In order to clarify this, all affected independent claims have been amended to recite that the user interface and controller comprises a programming environment having a plurality of programming functionalities.

The Examiner has also rejected claims 1-53, 59-64 and 103 under 35 U.S.C. §112, second paragraph as being indefinite.

In particular, the Examiner asserts that it is not clear whether just the controller or both the controller and the user interface are modified by the phrase "having a plurality of programming functionalities".

The user interface and controller is the programming environment referred to in the specification at page 15 which has the plurality of programming functionalities. As discussed above, the full programming environment 10 shown in Fig. 1 is synonymous with the "user interface and controller" and therefore it is the "user interface and controller" which has the plurality of programming functionalities. In view of the amendment to recite that the user interface and controller comprises a programming environment" it is submitted that this rejection also should be withdrawn.

With respect to claims 53 and 103, these claims refer to the ability of the system of the invention to insert text into the document at all locations in the document affected by the information solicited by the sequence or requests. The text that is inserted can be changed in the various locations. The text is dynamically alterable. Accordingly, text is different in at least one location from the other locations in which the text is inserted. To clarify this, the claims have been amended to recite that said text, i.e., the inserted text, is changed in at least one location from the text inserted in the other locations. Similar amendments have been made to claim 103.

In view of the above, it is submitted that all rejections under 35 U.S.C. §112, first and second paragraphs should be withdrawn.

Turning to the rejections based on the prior art, the Examiner has rejected claims 1-15, 17-22, 24-29, 31, 38-44, 48, 50-51, 53-60, 62-81, 88, 90-95, 98, 100-101, 103-110 and 112-115 under 35 U.S.C. §103 as being unpatentable over U.S. Patent No. 6,314,415 to Mukherjee in view of Capsoft, Inc., "HotDocs 5 Features" provided in Applicant's information disclosure statement.

Claim 16 is rejected under 35 U.S.C. §103 as being unpatentable over Mukherjee in view of HotDocs and further in view of Microsoft Corp, Microsoft Visual Basic Programmer's Guide.

Claim 23 is rejected under 35 U.S.C. §103 as being unpatentable over Mukherjee in view of HotDocs and IQDOC, LLC, Intelligent Document Assembly.

Claims 30, 32-34, 52, 82-85 and 102 are rejected under 35 U.S.C. §103 as being unpatentable over Mukherjee in view of HotDocs and further in view of U.S. Patent No. 6,473,892 to Porter.

Claims 35-37, 47, 61, 86-87, 89, 97 and 111 are rejected under 35 U.S.C. §103 as being unpatentable over Mukherjee in view of HotDocs, Porter and U.S. Patent 5,792,751 to Schoolcraft.

Claims 45-46, 49, 96 and 99 are rejected under 35 U.S.C. §103 as being unpatentable over Mukherjee in view of HotDocs and further in view of U.S. Patent No. 5,367,619 to Dipaolo et al.

All rejections of all claims rely on Mukherjee and HotDocs.

The present invention according to claim 1, relates to a system for automated drafting of a customized document, the system interfacing with a word processing application having a plurality of word processing functionalities and with a data base having a plurality of data base functionalities. The system comprises a user interface and controller comprising a programming environment having a plurality of programming functionalities. The user interface and controller interfaces with the user by presenting a sequence of requests to the user so that the user provides information necessary to prepare the document. The user interface and controller further communicates with the data base whereby the data base supplies first and second data stored in the data base to the user interface and controller, the first data being provided to the user interface and controller for providing a format for the requests to the user and the second data being provided to the user interface and controller for preparing text in the document.

The user interface and controller and the word processing application communicate so that the user interface and controller controls the word processing application to prepare and generate the document.

Furthermore, as now amended, the claim recites that the user interface and controller comprises a program for providing the sequence of requests to the user to obtain the information from the user for the preparation of the document and further for manipulating the document using the word processing application in response to the information received from the user and the second data from the data base to draft the contents of the document and further wherein the sequence of requests is dynamically altered by the program in response to the information provided in response to previous requests, the program determining the next request of the dynamically alterable sequence of requests to be provided to the user and when and how to manipulate the document to draft the contents of the document, the sequence of requests being such that each request is a logical extension of information provided by the user in response to previous requests and each request that is not the logical extension of information provided by the user in response to previous requests is omitted.

According to the present invention, the system claimed has a user interface and controller which provides the sequence of requests to the user to obtain the information from the user for the preparation of the document and for manipulating the document using the word processing

application in response to the information received from the user and the second data from the data base to draft the contents of the document. Thus, the system of the present invention manipulates the document using the word processing application to actually draft the contents of the document. Further, the sequence of requests is dynamically altered in response to information provided in previous requests, and furthermore, the program determines when and how to manipulate the document to draft the contents of the document in response to the information provided.

Applicant has reviewed the Mukherjee reference taken in context with the HotDocs document. As Applicant has explained in the "Background of the Invention" portion of the specification, the HotDocs program allows a user to create a program to interface with a word processing program to manipulate a document but HotDocs does not take full advantage of a programming environment to control a word processing environment and a data base to manipulate a document to actually draft the contents of the document. Programs such as HotDocs are essentially field population programs. They allow the user to populate fields of a form document, but they do not allow actual drafting of the contents of the document. The present invention, because of its ability to utilize all the programming functionalities of the programming environment, does more than merely populate fields. When information is provided by the user in response to the series of requests, the document is actually being drafted so that information provided by the user is used to draft the contents of the document.

The program desccribed in the Mukherjee reference cited by the Examiner is essentially an enhanced version of the HotDocs program. The Examiner has cited two portions of the Mukherjee reference which discuss of dynamically representing information based on previous responses including steps of displaying a first set of information for which data selection is required and using the first set of information to fire rules in an inference engine wherein the rules produce conclusions that are used to dynamically generate a second set of information for which data selection is required. See col. 2, lines 51-58 of Mukherjee. However, Mukherjee is also a field population program. It is a program for automated forms publishing and it is capable of obtaining information from the user for populating fields of the forms being published by the program. However, the Mukherjee program is not capable of actually using the information to

draft the contents of the document itself. Mukherjee starts with forms which are populated in response to information provided by the user. Mukherjee is incapable of manipulating the document using the word processing application in response to the information received from the user and the second data from the data base to draft the contents of the document. Further, even when taken in combination with HotDocs, Mukherjee is incapable of manipulating the document to draft the contents of the document.

In contrast to the invention, Mukherjee allows the user to provide information in response to fields that appear on the display. When a user enters information through the computer display, various rules in the data base will be "fired" thus changing the appearance of the display for future prompts. The dynamically changing nature of the graphical-user interface avoids prompting the user for redundant or irrelevant information. See col. 5, lines 40-45. A display generator generates screens such as those shown in Figs. 3A-3L using well known "Windows" display techniques. See col. 5, lines 56-60. After a user has entered information in response to the dyamically changing prompts, new forms can be printed. However, Mukherjee does not draft the document itself but populates the fields of a form document with the information requested. In contrast, the present invention creates a document from a blank page. The present invention could be used to draft a 100 page document from a blank page. Instead, Mukherjee relies upon an existing document (form) and populates the fields of that document with the information requested.

Furthermore, there is an additional difference between programs such as those of Mukherjee and HotDocs and that of the present invention. Mukherjee uses a limited set of expert rules which are "fired". The rules determine which user interface features to display at a particular point in the data entry sequence. See col. 2, lines 42-44 of Mukherjee. Thus, Mukherjee's program relies upon a limited set of rules which are followed in response to information provided by the user. In contrast, the present invention takes full advantage of the programming environment to control a word processing environment to manipulate and draft a document. The prior art systems are not capable of efficiently preparing a complex document because they do not have the ability to utilize all the functionalities of a full programming environment and data base. Mukherjee relies upon a limited set of expert rules in order to

populate the fields of a form. The present invention in contrast, uses all logical functions and/or combinations of logical functions and/or embedded such logical functions of a full programming environment at all stages of document preparation to generate the desired document. Mukherjee does not utilize a programming environment and all its functionalities to control a word processing environment to draft a document. Rather, Mukherjee uses a limited set of rules contained within an "expert system rules and facts" database 128 to frame the requests to be put to the user to populate a form document. Mukherjee can only use the rulers in his database. The prior art systems like Mukherjee, including Hot Docs, are not capable of using combinations of logical functions or using using such logical functions in a manner such that logical functions are nested within each other. Thus, Mukherjee adds nothing to the Hot Docs program that would result in the invention. The present invention is capable of using combination of such programming functionalities, which allows the full advantage of using a full programming environment to control a full word processing environment to be achieved. See page 2, lines 13-27 of the present application.

In order to highlight this distinction, the claims have been amended to recite that the user interface and controller uses "combinations of the programming functionalities to prepare and generate the document."

In the present invention, unlike other document generation systems, the user interfaces with a full programming environment 10 programmed in a language such as Visual Basic to prepare a document which is generated by a full word processing environment 20 responding to instructions from the full programming environment 10 and data from data base environment 30. This provides significant advantages over prior art systems. In particular, this allows use of any and all logical functions, either alone, or in combination, and access to all the needed functionalities of the full word processing environment 20 while at the same time allowing access to all the needed functionalitis of the full programming environment 10 to generate the document. This allows the system to dynamically structure the way in which information is obtained for preparing the document, that is, for drafting the contents of the document. See page 17, lines 1-11 of the specification.

The full programming environment 10 includes all the functionalities provided in the programming language selected, in the embodiment described, Visual Basic. Full programming environment 10 provides instructions to the full word processing environment 20 which may be Microsoft Word, installed on the computer 1. The full programming environment 10 sends instructions and data to the full data base environment 30. The full word processing environment includes a plurality of well known word processing functionalities. The programming environment 10 includes a plurality of programming environment functionalities which include, without limitation, mathematical functions, verifying matches, graphical user interfaces and logical functions such as SELECT CASE, IF THEN, loops, etc. or combinations of these. See, p.15, lines 22-25. Word processing environment functionalties include AUTO TEXT, book marks, fields, tables, etc. There is also a full data base environment which provides responsive data from the data base to the full programming environment 10. This data comprises two types of data. The first type of data is used to populate screen display forms so that appropriate questions are presented to the user to solicit information for the document. The second type comprises data stored in the forms for insertion in the document, for example, alternate provisions for insertion in the document.

By interfacing with the full programming environment, the user can prepare a document generated by the full word processing environment, using any and all logical functions or combinations of these functions of the word processing environment and the programming environment to generate the document. The system of the present invention automatically drafts the document. In contrast, programs such as Mukherjee's program and Hot Docs allow forms to be populated but do not allow a document to be drafted from scratch.

The present invention can draft a document from scratch because it is not based upon a limited set of expert rules, as is Mukherjee's program or as is the HotDocs program. Instead, it uses a full programming environment and a full word processing environment to actually draft the contents of a document.

Accordingly, it is submitted that the Mukherjee reference, taken alone or in combination with the HotDocs reference, does not teach or suggest the present invention as now claimed.

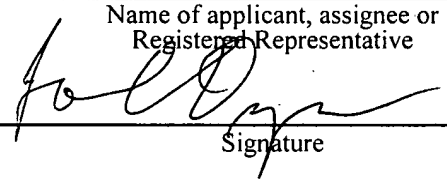
Applicant has amended all independent claims in this application to recite that the user interface and controller comprising a programming environment obtains the information from the user to manipulate the document using the word processing application in response to the information received from the user to draft the contents of the document. The Mukherjee and HotDocs programs are incapable of drafting the contents of a document, that is, automatically creating a document. They are merely programs for populating fields in an existing form. As such, they do not teach or suggest the present invention, taken alone or in any combination.

Because all rejections rely on both Mukherjee and HotDos, it is submitted that all rejections of the claims should be withdrawn in view of the above amendments and remarks. Applicant has considered the other references cited and applied by the Examiner but submits that they fail, alone or in combination, to teach or suggest the invention now claimed.

In view of the above, Applicant submits that all claims in this application are now in condition for allowance, prompt notification of which is requested.

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on December 22, 2003:

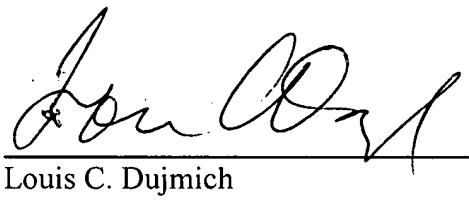
Louis C. Dujmich
Name of applicant, assignee or Registered Representative



Signature

December 22, 2003
Date of Signature

Respectfully submitted,



Louis C. Dujmich
Registration No.: 30,625
OSTROLENK, FABER, GERB & SOFFEN, LLP
1180 Avenue of the Americas
New York, New York 10036-8403
Telephone: (212) 382-0700

LCD:cfm